The Evolution of Blockchain Security

PRESENTER

Jutta Steiner, CEO & Co-Founder, Parity Technologies

PhD Mathematics, University of Bonn

Ex-Chief Security, Ethereum Foundation

Guardian (Advisor), matrix.org Foundation

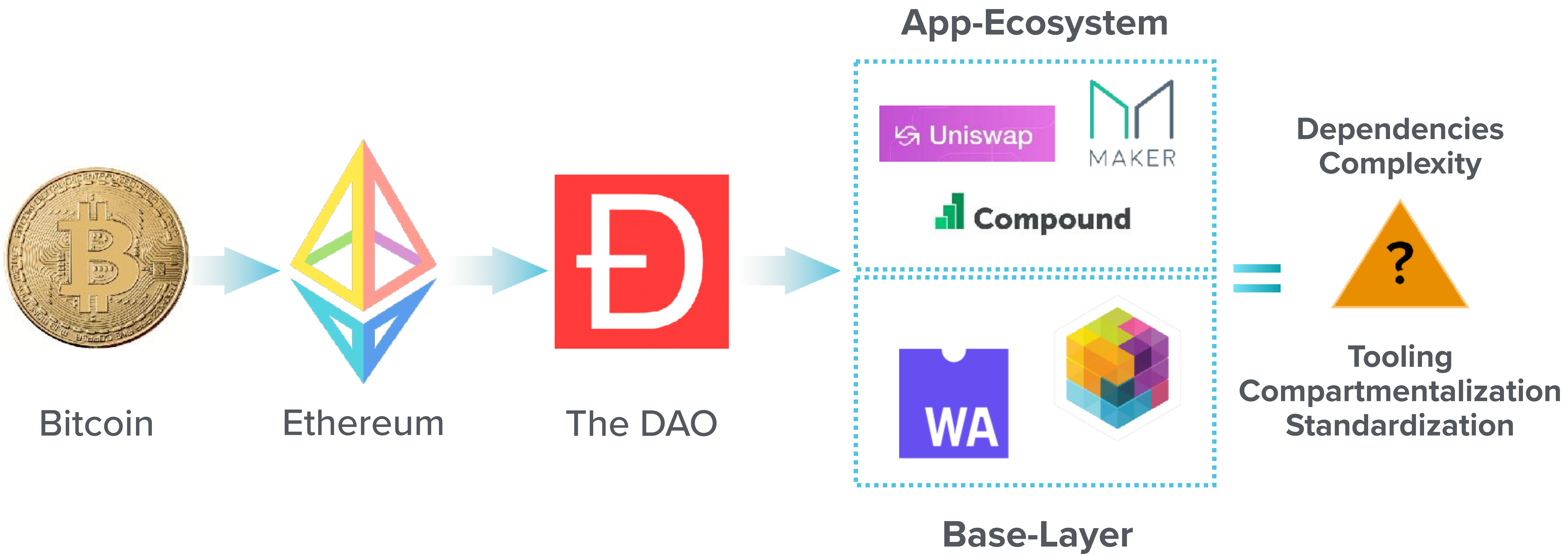
Important Disclosures

The views expressed here are those of the individual AH Capital Management, L.L.C. (“a16z”) personnel quoted and are not the views of a16z or its affiliates. Certain information contained in here has been obtained from third-party sources, including from portfolio companies of funds managed by a16z. While taken from sources believed to be reliable, a16z has not independently verified such information and makes no representations about the enduring accuracy of the information or its appropriateness for a given situation.

This content is provided for informational purposes only, and should not be relied upon as legal, business, investment, or tax advice. You should consult your own advisers as to those matters. References to any securities, digital assets, tokens, and/or cryptocurrencies are for illustrative purposes only and do not constitute a recommendation to invest in any such instrument nor do such references constitute an offer to provide investment advisory services. Furthermore, this content is not directed at nor intended for use by any investors or prospective investors, and may not under any circumstances be relied upon when making a decision to invest in any fund managed by a16z. (An offering to invest in an a16z fund will be made only by the private placement memorandum, subscription agreement, and other relevant documentation of any such fund and should be read in their entirety.) Any investments or portfolio companies mentioned, referred to, or described are not representative of all investments in vehicles managed by a16z, and there can be no assurance that the investments will be profitable or that other investments made in the future will have similar characteristics or results. A list of investments made by funds managed by Andreessen Horowitz (excluding investments for which the issuer has not provided permission for a16z to disclose publicly as well as unannounced investments in publicly traded digital assets) is available at https://a16z.com/investments/.

Charts and graphs provided within are for informational purposes solely and should not be relied upon when making any investment decision. Past performance is not indicative of future results. The content speaks only as of the date indicated. Any projections, estimates, forecasts, targets, prospects, and/or opinions expressed in these materials are subject to change without notice and may differ or be contrary to opinions expressed by others. Please see https://a16z.com/disclosures for additional important information.

We’ve Come a Long Way…



Bitcoin

Ethereum

The DAO

App-Ecosystem

Base-Layer

Dependencies
Complexity

Tooling
Compartmentalization
Standardization

Any Andreessen Horowitz investments and portfolio companies described or referred to above are not representative of all investments in vehicles managed by the firm and there can be no assurance that the investments described are or will be profitable or that other investments made in the future will have similar character or results. See Important Disclosures at the outset for more information

SECURITY IS NOT JUST CODE!

“Weakness in an information system, system security procedures, internal controls, or implementation that could be exploited or triggered by a threat source.”

- NIST Vulnerability definition

CONSIDERATIONS FOR SMART CONTRACTS DEVELOPMENT

What Can Go Wrong with Code, and How to Mitigate

ISSUE	EXAMPLE
Memory safety	Overflows, underflows, dangling pointers
Input validation	Code injection, format string hacks, sql injection, etc.
Privilege escalation flaw	Access controls
Fundamental design flaws	Denial of Service (DoS)
Side channel attacks	Timing attacks
Cryptographic vulnerabilities	Insecure key storage, randomness of keys

MITIGATION

- Threat modelling
- Audits
- Testing
- Fuzzing

Secure Smart Contract Code!?

LEARNINGS

Frequency and nature of vulnerabilities for smart contract code and normal code is similar, but:

- What you read about does not necessarily equate to what you should be worried about
- A lot of the findings (almost 49%) are almost impossible to imagine detecting with a tool or testing

Smart contract development is the opposite from agile!

ToB Report: https://blog.trailofbits.com/2019/08/08/246-findings-from-our-smart-contract-audits-an-executive-summary/

A Comprehensive Checklist for Smart Contract Development

PARITY TECHNOLOGIES
14 POINT CHECK LIST

https://www.parity.io/paritys-checklist-for-secure-smart-contract-development/

Highlights from the Check List

GitHub and Repo Structure

- Create a new GitHub organization
- Put every contract in a separate repo
- Embed dependencies

Deployment

- Actual deployed state of each contract should live in a protected `master` branch
- Every contract should have a README that lists its deployment addresses in all networks

Code Quality

- Make sure that bugs related to syntax quirks and misunderstandings are discoverable with tests by using a different language
- Reviews should be required for pull requests

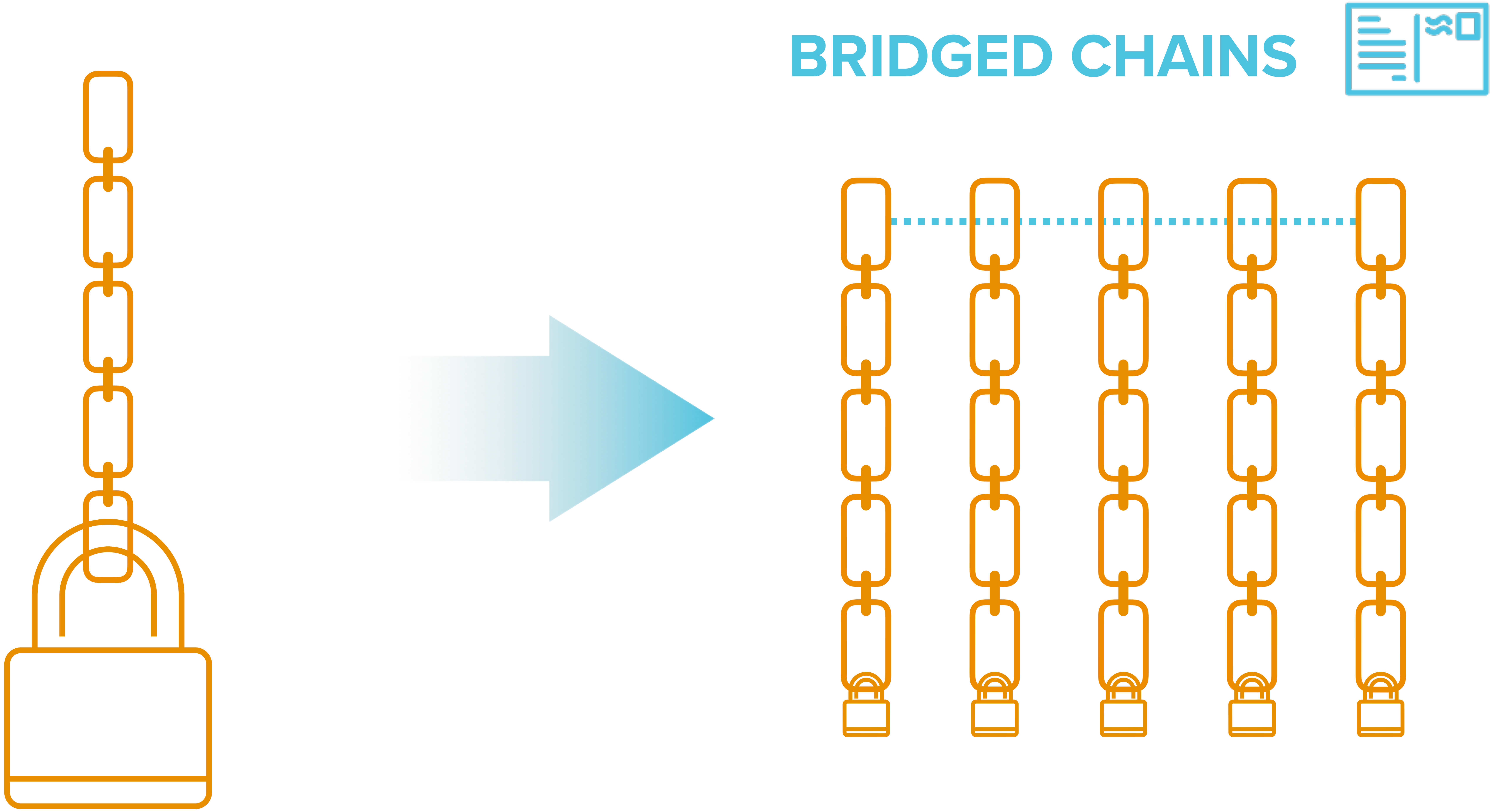
Beyond Code: Security in a Developing Interdependent and Open Ecosystem

SOME OBSERVATIONS

- More and more projects are rolling their own chains vs. “Don’t roll your own crypto!”
- Limitations in scalability: Chains are competing for security
- Limitations in framework: App ecosystem is developing complex interdependencies

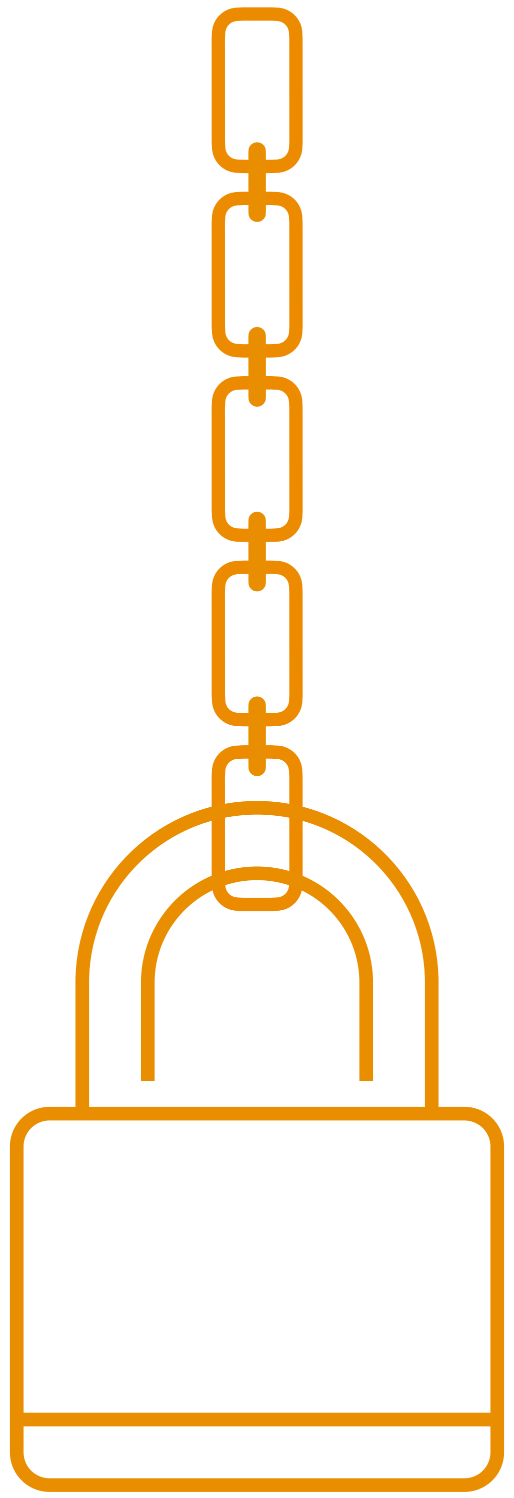
SOLUTIONS AND CONSIDERATIONS GOING FORWARD

Naive Scaling:
Fractured Security and Weak Interoperability



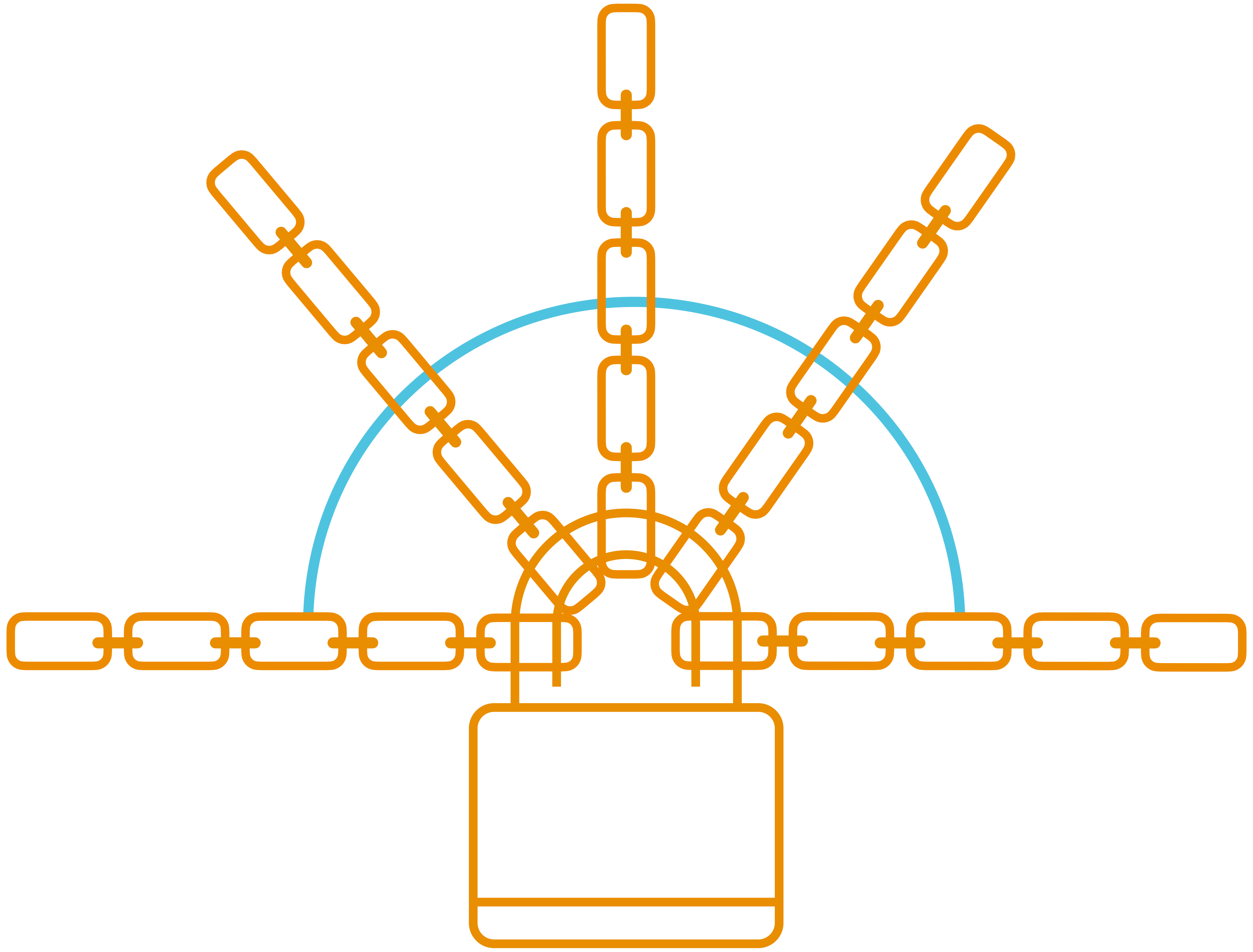
Better Scaling: Pooled Security and Strong Interoperability

SHARDED BLOCKCHAIN



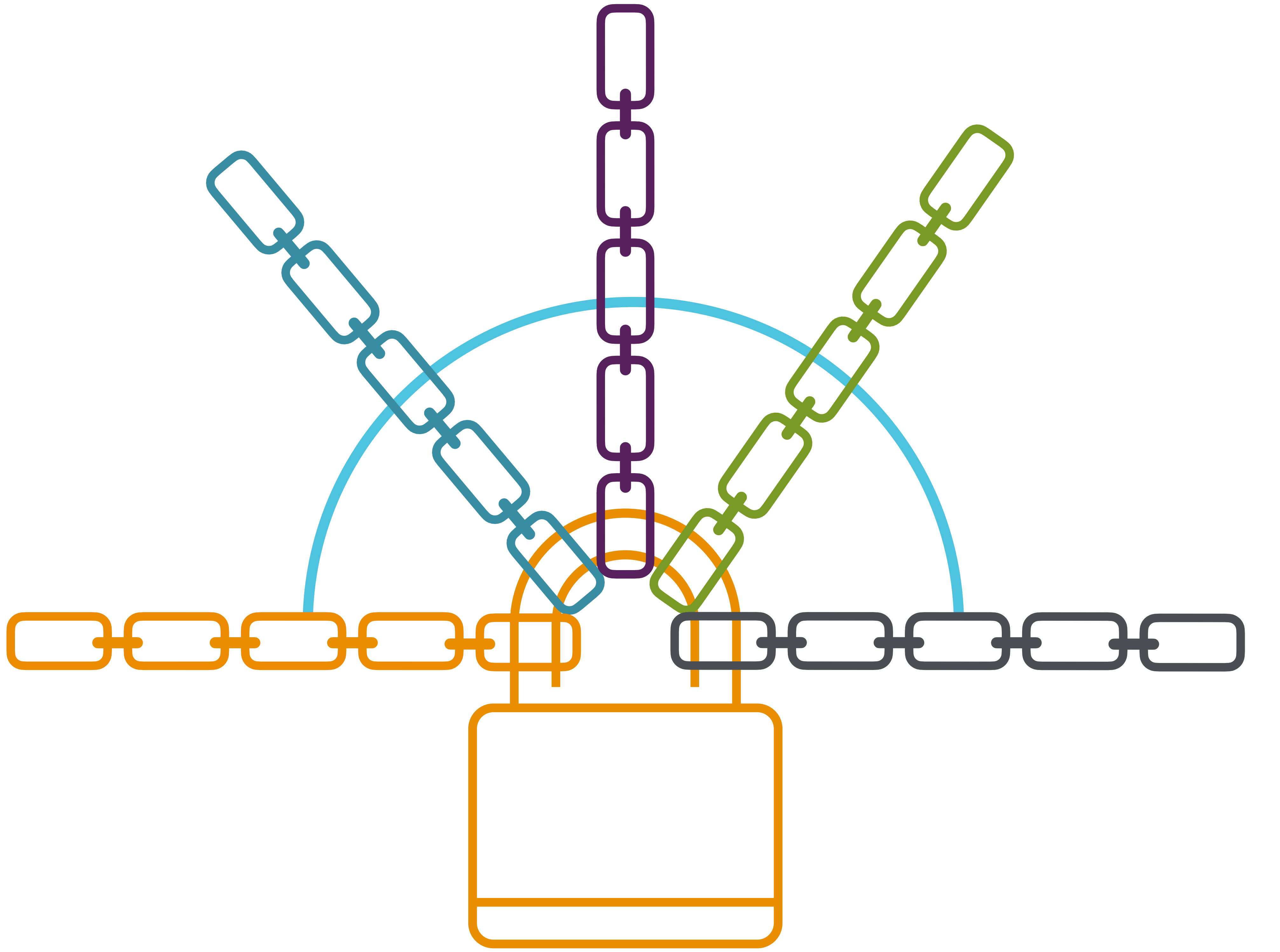
Moving On from a One-size-fits-all Approach...

SHARDED BLOCKCHAIN

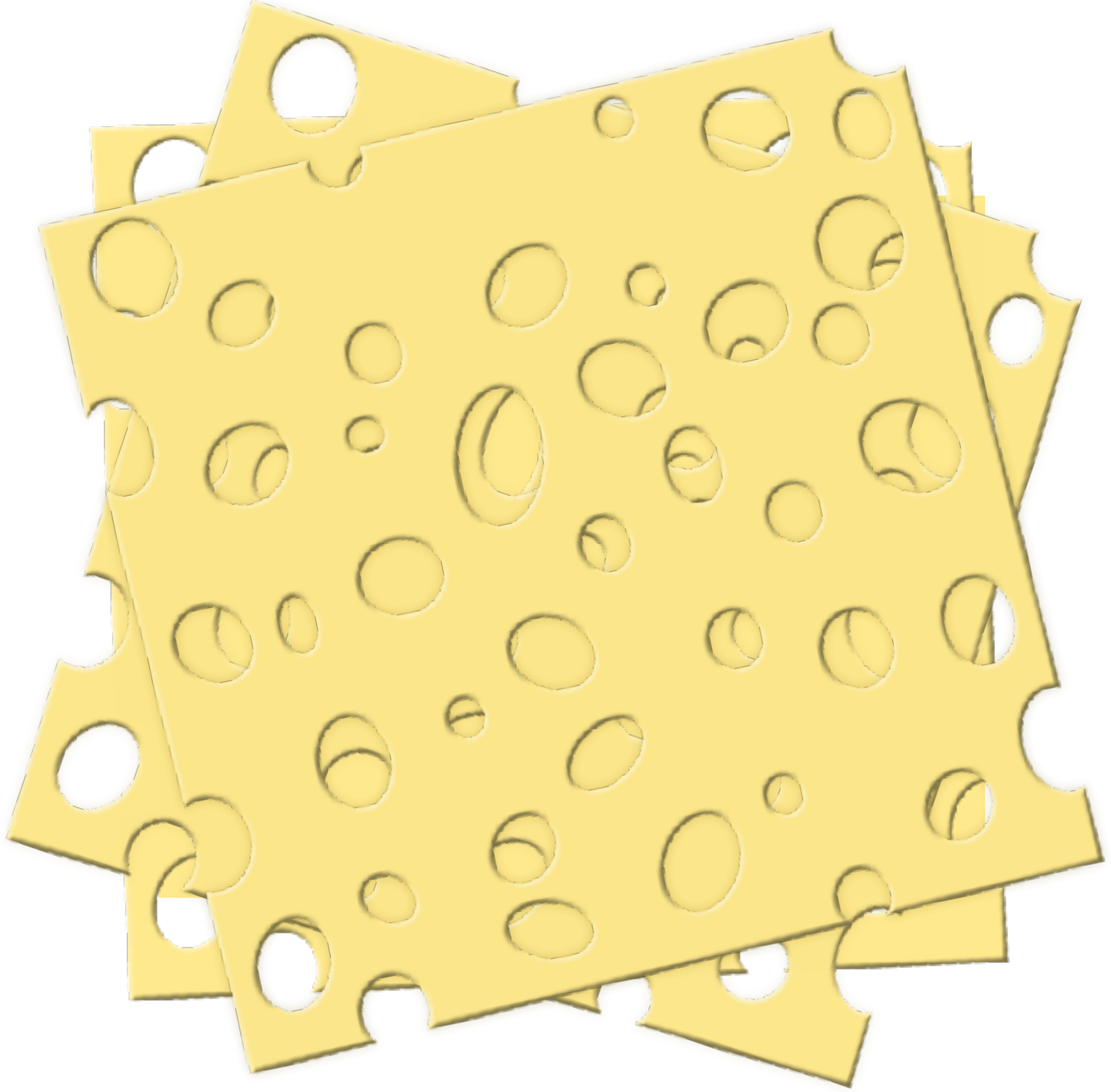


Achieving Customization and Compartmentalization

HETEROGENOUS MULTICHAIN



Build a Structured Framework to Ease Development and Close Security Holes



Customizable runtime models vs. one-size-fits-all Turing complete virtual machines

Resort to standards like Wasm and “safer” languages like Rust

On-chain governance in case of ultimate failure

What Blockchain Can Learn from Other Industries

Aerospace

Medicine

Hardware

Communication

Open Source

Key Takeaways

Security is more than code

Smart contracts aren’t secure

Don’t roll your own blockchain

Be humble and learn from other industries

Security is hard and we’re in this together a16z

[CRYPTO STARTUP SCHOOL]